



SYSTEM AND METHOD FOR COMMUNICATING INFORMATION RELATING TO A NETWORK RESOURCE

CROSS REFERENCE TO RELATED APPLICATIONS

This application is a continuation-in-part of copending application Serial No. 09/903,923 (entitled "A System and Method for Communicating Information Relating to a Network Resource"), filed July 12, 2001, which is a continuation of Serial No. 09/193,756 (entitled "A System and Method for Communicating Information Relating to a Network Resource"), now U.S. Patent No. 6,292,813, which claims the benefit of provisional application serial no. 60/066,058 (entitled "A System and Method for Communicating Information Relating to a Network Resource"), filed November 17, 1997, now expired.

This application is also a continuation-in-part of copending non-provisional application Serial No. 10/403,936 (entitled "Communication of Information Relating to a Network Resource"), filed March 31, 2003, which is a continuation-in-part of non-provisional application serial no. 09/566,703 (entitled "Communication of Supplemental Information For A Network Resource"), filed May 9, 2000, which claims the benefit of provisional application serial nos. 60/172,168 (entitled "Communication of Supplemental Information For A Network Resource"), filed December 17, 1999, 60/167,479 (entitled "Communication of Supplemental Information For A Network Resource"), filed November 24, 1999, and 60/133,781 (entitled "Communication of Supplemental Information For A Network Resource "), filed May 12, 1999.

Serial no. 10/403,936 is also a continuation-in-part of serial no. 09/258,873 (entitled "Communication of Information Relating to a Network Resource"), filed February 26, 1999, which claims the benefit of provisional application serial nos. 60/110,776 (entitled "Communication of Information Relating to a Network Resource"), filed December 3, 1998, 60/091,708 (entitled "Communication of Information Relating to a Network Resource"), filed July 3, 1998, and 60/076,147 (entitled "Communication of Information Relating to a Network Resource"), filed February 26, 1998. All of the applications identified above are hereby incorporated by reference into this application.

BACKGROUND AND SUMMARY

A continuing difficulty in today's information-rich society is the effective communication of pertinent information to individuals who are most interested in such information. Some of the systems and methods disclosed herein relate to the fields of information retrieval and information presentation. Specific embodiments and features, and the nature of the present invention may be understood by reference to the following detailed description and the drawings.

BRIEF DESCRIPTION OF DRAWINGS

Figure 1 is a block diagram of a computer system attached to a network.

Figure 2 is a diagram of multiple computers on a network.

Figure 3 is a simplified representation of a screen showing a window in which information retrieved over a network is displayed.

Figure 4 is the view of Figure 3 with an additional window shown on the display, in which supplemental information is displayed.

Figure 5 is a network diagram showing the retrieval of supplemental information by a client computer.

Figure 6 is a flow chart illustrating the process for retrieving and presenting supplemental information in the network of Figure 5.

Figure 7 is a network diagram showing an alternate procedure for retrieval of supplemental information.

Figure 8 is a flow chart illustrating the process for retrieving and presenting supplemental information in the network of Figure 7.

Figure 9 is a network diagram showing an alternate procedure for retrieval of supplemental information.

Figure 10 is a flow chart illustrating the process for retrieving and presenting supplemental information in the network of Figure 9.

Figure 11 is a network diagram showing network activity and inter-process communication at the client computer.

Figure 12 is a flow chart illustrating the process diagrammed in Figure 11.

Figure 13 is a network diagram showing an alternate implementation of two processes executing at the client program.

Figure 14 is a flow chart illustrating the process diagrammed in Figure 13.

Figure 15 is a flow chart illustrating a server process similar to that shown in Figures 11 and 13.

Figure 16 is a simplified representation of a database or look-up table that may be used by a server computer.

Figure 17 is a flow chart illustrating a process in which a client computer sends a request to a reference server in certain circumstances.

Figure 18 is a flow chart illustrating a process in which a client computer determines whether supplemental information is available for more than one document.

Figure 19 is a view of a screen showing a window in which supplemental information is displayed.

Figure 20 is a view of a desktop having a display element that the user can select to display supplemental information.

Figure 21 is the desktop of Figure 20 after selection of the display element.

Figure 22 illustrates a display screen on which supplemental information is scrolled in a window.

Figure 23 illustrates a display screen informing the user that supplemental information or supplemental content relating to other content on the display screen exists.

Figure 24 illustrates the display screen of Figure 23 when supplemental information is presented to the user.

Figure 25 illustrates the display screen of Figure 24 after the user selects the supplemental information presented to the user in Figure 24.

Figure 26a illustrates rotation of supplemental information in a window in Figure 24.

Figure 26b is a flow chart of a process in which the displayed supplemental information remains substantially constant during the time that the input device is active.

Figure 27 illustrates a browser or document viewing program having an area in which supplemental information may be displayed.

Figure 28 is a flow chart illustrating a process for presenting a supplemental resource or supplemental content to a user.

Figure 29 illustrates the selection of a display element while content in a browser is presented to a user.

Figure 30 illustrates the updated display screen after the selection of the display element in Figure 29.

Figure 31 illustrates the display screen of Figure 30 showing display elements representing various types of supplemental information on a task bar.

Figure 32 illustrates the updated display screen upon selection of a task bar display element in Figure 32.

Figure 33 is a network diagram illustrating network activity in connection with Figures 29, 30, 31, and 32.

Figure 34 illustrates a number of display elements that may be used to represent different types of supplemental information.

Figure 35 is a flow chart of a process similar to that illustrated in Figures 29 to 33.

Figure 36 is a flow chart illustrating an alternate process.

Figure 37 is a flow chart of the process illustrated in Figures 38a and 38b.

Figure 38a illustrates information being displayed in a window.

Figure 38b illustrates supplemental information being displayed in the window of Figure 38a.

Figure 39 illustrates an alternate method of displaying supplemental information.

Figure 40 is a network diagram in which supplemental information from more than one source is retrieved.

Figure 41 illustrates a menu being displayed upon selection of a display element.

Figure 42 illustrates presentation of supplemental information upon selection of a menu item in Figure 41.

Figure 43 illustrates an alternative menu being displayed upon selection of a display element.

Figures 44 to 46 illustrate another procedure for selection of the desired type of supplemental information.

Figure 47 illustrates an alternate way of presenting supplemental information in connection with the procedure illustrated in Figures 44 to 46.

Figures 48 and 49 illustrate the selection and display of supplemental information relating to audio content.

Figure 50 illustrates the presentation of the audio content relating to the supplemental information of Figure 49.

Figure 51 and Figure 52 are diagrams illustrating various procedures for installing and executing software.

Figure 53 and Figure 54 are flow charts illustrating procedures for installing and executing software.

Figure 55 is a diagram of the relationships between six documents.

Figures 56a, 56b, 56c, 56d, 56e, and 56f are representations of the six documents from Figure 55.

Figure 57 is a representation of four search documents and three related documents.

Figure 58 is a representation of four search documents and three related documents with a display view and one anticipated view.

Figures 59(a) and 59(b) are each a representation of four search documents and three related documents showing a display view and four anticipated views.

Figures 60(a) and 60(b) are each a representation of four search documents and three related documents showing various term views.

Figure 61 is a representation of four search documents and three related documents showing various subdocument views.

Figure 62 shows seven documents ordered according to four different ordering characteristics.

Figure 63 is a flow chart of the operation of a database system in one embodiment of the present invention.

Figure 64 is a flow chart of the operation of a database system in an alternate embodiment of the present invention.

Figure 65 is a flow chart of the operation of a database system in an alternate embodiment of the present invention.

DETAILED DESCRIPTION

Figure 1 is a block diagram of a general purpose computer 102 that may be used to implement one or more embodiments of the present invention. The computer 102 has a central processing unit (CPU) 104, memory 113, and input/output (i/o) circuitry 112. The CPU 104 is connected to the memory 113 and the i/o circuitry 112. The i/o circuitry permits the CPU 104 to access various peripheral devices, such as the display 108, local storage 106, and input device(s) 110. The input device(s) 110 may include a keyboard, mouse, pen, voice-recognition circuitry and/or software, or any other input device. Some type of secondary or mass storage 106 is generally used, and could be, for example, a hard disk or optical drive. The storage 106 can also be eliminated by providing a sufficient amount of memory 113. Either the storage 106 or the memory 113 could act as a program storage medium that holds instructions or source code. The i/o circuitry 112 is also connected to a network 114, thereby connecting the computer 102 to other computers or devices.

Figure 2 is a representation of multiple computers (251, 252, 253, 254, 255, 256, and 257) connected together to form a network of computers and/or networks. Computers 251, 252, and 256 are shown connected to wide area network (WAN) 263, whereas computers 253, 254, 255, and 257 are shown interconnected by local area network (LAN) 261. The LAN 261 is connected to the WAN 263 by connection 262.

Figure 3 shows a simplified representation of a video display screen 304 for a computer such as that of Figure 1. The area 304 represents the area on a screen within which images, text, video, and other type of data or multimedia objects can be displayed and manipulated. On the display 304 shown in Figure 3, a cursor 301, and a number of display elements, icons, or objects 302 are shown.

Another type of object, window 306, is also displayed on the display 304. The window 306 is a representation of a document retrieval, browsing, and/or viewing program that is used to view or interact with information or resources either stored locally on the computer or retrieved over a network. The window 306 has a title area 308 that displays the title of the resource or document being displayed, along with the server on which the document is located. The title area 308 could also display the location or address of the document being displayed, or also the universal resource locator of the document being displayed. Alternatively, an additional area within the window could

be used for displaying the universal resource locator in the manner implemented by commercially available browsers.

Shown within the document viewing area of the window 306 in Figure 3 are the contents of a document or page published by Universal Theatres, Inc., and made available at its server over a wide area or global network (e.g., the Internet's World Wide Web). This particular document or web page relates a motion picture produced by Universal Theatres entitled "Flying Over The Precipice." As indicated in the window 306, actors named Warren Barrett and Annette Foster star in the movie. The document or page shown in window 306 is made available by Universal Theatres (on a wide area or global network) for browsing by, for example, people that may be interested in seeing this particular movie. The page in Figure 3 (or others accessible from it) may provide information about which local theaters are showing the movie, and the times it is being shown. The page may also offer previews of the movie in the form of audio or video clips that encourage the viewer to see the movie.

Still referring to Figure 3, some promotional information is also provided in an area 310 on the right side of the page. This promotional information is in the form of a list of favorable reviews that have been given to the "Flying Over the Precipice" movie. For example, Joel Seagull of ABC News gave the movie "five stars," and other reviewers also gave the movie a favorable review. These reviews might be selected by Universal Theatres, Inc. for inclusion in this page to encourage the viewers of the page to see the movie. For this reason, favorable reviews may sometimes be the only reviews included such a page.

A banner advertisement 312 is also displayed at the top of the page, which promotes another Universal Theatres movie, "Time Will Tell." Selection of such a banner ad may retrieve and display in the window 306 information about the advertised movie. The banner ad 312 could alternatively promote other entertainment choices, or other goods or services if Universal Theatres chooses to sell advertisement space on its pages to third parties. For example, a supplier of soft drinks to patrons of movie houses showing Universal Theatres pictures may choose to invest in advertising on one or more of Universal Theatres' pages.

In Figure 4, a simplified representation of a video display or screen is shown where window 306 is present, as was the case in Figure 3, but in addition, a window 320 is also shown on the display. In the embodiment shown, the window 320 is displayed automatically when the user

retrieves and displays the Universal Theatres page shown in window 306. The window 320 contains excerpts from various reviews of the movie “Flying Over the Precipice.”

In the particular embodiment shown in Figure 4, the information in the window 320 is not created or published by Universal Theatres. This information is instead created and published by a third party in this embodiment, such as a movie review organization that seeks to provide a well-balanced, and even-handed representation of the reviews a particular movie is receiving. The page published by Universal Theatres and shown in window 306 in Figure 4 may not link to or point to where the information in window 320 can be found on the network.

Shown in window 320 are four reviews of the movie “Flying Over the Precipice,” including the favorable review by Joel Seagull of the (fictional) ABC News. The other three reviews shown in window 320 (from Wire Services Ratings, MetroPages Reviews, and TimeLine Reviews) are not as favorable and may discourage the viewer from seeing the movie “Flying Over the Precipice.” In fact, the information provided in window 320 reveals quite a different tone from the reviewers than the set of reviews cited by Universal Theatres on its “Flying Over the Precipice” web page in window 306. Universal Theatres has omitted from its “Flying Over the Precipice” web page (shown in window 306) some of the reviews quoted in window 320.

As can be seen from the embodiment of Figure 4, the information provided by Universal Theatres on the page shown in window 306 may not be an accurate or even-handed representation of the reviews the subject movie is receiving. Nor would one necessarily expect that Universal Theatres would provide an entirely accurate representation of the reviews for Flying Over the Precipice if many of those reviews are unfavorable. To do so may be counterproductive for Universal Theatres, because unfavorable reviews would not effectively sell the movie to people who view the page shown in window 306. The information provided by Universal Theatres in its publications and on its web pages may not be wrong or incorrect, but at least in some situations, this information is likely to be slanted in a manner that is favorable to Universal Theatres.

In the embodiment shown in Figure 4, the information provided in window 320 provides a revealing supplement to the information on the Universal Theatres page shown in window 306. The information in window 320 therefore may be quite useful to a person contemplating going to a movie because the viewer will be able to consider a more representative sample of the reviews the “Flying Over the Precipice” is receiving. Universal Theatres, however, may prefer to keep

unfavorable reviews from the viewer of its page when the reviews are generally unfavorable. (But in other situations or embodiments, Universal Theatres may not object, and may even encourage the communication of information of this type to the user.)

It may be possible for the user to go directly to the place on the network where the information shown in Figure 320 is found. However, at least in some situations, the user may not know where to find these reviews, and may not even know that there is a resource for such information. Yet the user may in these situations know where to find the Universal Theatres' "Flying Over the Precipice" page that is shown in window 306. Universal Theatres may have heavily marketed the movie, and as a result, the user may know the location on the network where information about the movie can be found. Thus, at least in some cases, the user is more likely to go to the Universal Theatres page when interested in a particular movie produced by Universal Theatres. Yet the information that the user will find there may describe Universal Theatres' movies favorably, making an assessment of the entertainment value of a particular movie both difficult and potentially inaccurate. The supplemental information shown in window 320 may make such an assessment easier or more accurate.

Figure 5 is a diagram showing seven computers, including computers 902, 904, and 906, connected to a network 910. Many more (or less) computers also could be present on the network. The network 910 could also be an internetwork such as that shown in Figure 2, or any other type of network. In the embodiment of Figure 5, the user operates a browsing program (i.e., "browser") at computer 902 (referred to herein as the "client" computer), and the browsing program is typically stored and executed on that computer. The browser is capable of retrieving resources from various computers on the network, such as from computers 904 and 906. Normally, inter-computer communication takes place through a protocol such as HTTP, DCOM, IIOP, or Java's RMI. Any other protocol could be used.

Figure 6 is a flow chart illustrating the process diagrammed in Figure 5. At 942 in Figure 6, the browser executing on computer 902 in Figure 5 sends a request to computer 904 for a document stored on server 904. The document stored on the server 904 might be the "Flying Over the Precipice" document shown in Figures 3 and 4, and the computer 904 could be a web server on which that document is stored. The request for this document is shown in Figure 5 by the arrow 921. The web server 904 responds to this request at 944 by sending the requested document to the

client computer 902. (See arrow 922 in Figure 5.) The browser executing at the client then displays the requested document (946).

At 948 in Figure 6, the browser (or another program that works in conjunction with the browser) then sends information describing and/or identifying the displayed document to computer 906 (arrow 923 in Figure 5). In the embodiment shown, the computer 906 differs from the server from which the browser retrieved the displayed document. A computer such as computer 906 may be referred to in some embodiments as a “reference server” because requests are sent to it to see if supplemental information is available for a particular document on the network. Thus, it is used, at least in part, as a “reference.” The information sent to computer 906 in the embodiment of Figure 5 includes the server from which the displayed document has been retrieved (i.e., server 904), as well as the identity of the document or resource on that particular server that is being displayed at the client 902. For the example of Figures 3 and 4, the server is the Universal Theatres, Inc. server and the specific resource on that server is the “Flying Over the Precipice” document. See title bar 308 in Figures 3 and 4.

Upon receipt of this information from the client computer 902, computer 906 checks a table or a database of servers and documents to determine if there is supplemental information available for this particular document. The computer 906 then sends a response to computer 902 that indicates whether supplemental information is available for the identified document (i.e., the “Flying Over the Precipice” document), and if so, the computer 906 includes the supplemental information in its response. (See arrow 924 in Figure 5.) The browser (or the program operating in conjunction with the browser) then displays the supplemental information (954). In the embodiment of Figure 4, the supplemental information available for the “Flying Over the Precipice” document stored on Universal Theatres’ web server is the reviews information shown in window 320 in Figure 4. This supplemental information is sent to the client computer 902 (arrow 924) by the computer 906, and is displayed in window 320 along with the “Flying Over the Precipice” document as shown in Figure 4.

Figure 7 is another network diagram, and Figure 8 is an accompanying flow chart. Computers 1002, 1004, and 1006, as well as others, are shown on a network 1010. The user operates the browsing program at the client computer 1002, and the browsing program is usually (but need not be) stored and executed on that computer. As in Figure 5, the browser is used to

retrieve resources from various computers on the network. At 1042 in Figure 8, the browser executing on the client 1002 sends a request to server 1004 for a document stored on resource server 1004 (see arrow 1021 in Figure 7). Upon receiving this request, the server 1004 sends the requested document to the client 1002 (arrow 1022 in Figure 7), and the document is displayed at the client 1002. At 1048 in Figure 8, the browser, or some other program that operates with the browser, sends information identifying the document displayed at the client to the server 1006. This computer 1006 checks to see if supplemental information is available for the identified document, and then sends to the client 1002 information that indicates whether supplemental information is available for the identified document, and if so, where that supplemental information can be found. If there is supplemental information available for the document displayed at the client computer 1002, the client then sends a request for the supplemental information to the computer identified by the information that the client receives from the reference server 1006. In the example of Figures 7 and 8, computer 1006 identifies computer 1008 as having supplemental information, so this request from computer 1002 goes to computer 1008 (see arrow 1025 in Figure 7). (The computer 1008 may be referred to in some embodiments as the “supplemental information” server.) The computer 1008 responds by sending the supplemental information to the client 1002 (arrow 1026), and the client presents this supplemental information to the user, typically by displaying the supplemental information on the screen.

Figures 9 and 10 illustrate a network diagram and accompanying flow chart in an other embodiment. In Figure 9, the browser at the client 1102 sends a request for a document to the server 1104, and this server responds by sending the requested document to the computer 1102 (arrows 1121 and 1122 in Figure 9). The client then sends information identifying the document to the reference server 1106 (arrow 1123). The reference server 1106 then determines, by accessing a database or look-up table, whether the identified document has supplemental information available for it (1152). If it does, it sends a message to computer 1108, directing that computer to send the supplemental information to the client 1102. The supplemental information is then sent to the client by the computer 1108. If the server 1106 were to determine at 1152 that there is no supplemental information available for the document, the server 1106 would respond to the client computer 1102 by informing the client computer 1102 that no supplemental information is available for the displayed document.

Figures 11 through 14 illustrate embodiments of the present invention where the client computer 1502 has more than one process executing on it. In one embodiment, one process is a browser 1520 or document display program, and the other process is a program, applet, thread, or the like (hereinafter “program 1525”) that coordinates the display of the supplemental information in the manner herein described. These two programs communicate through an interprocess communication protocol, such as the Dynamic Data Exchange (DDE), OLE, ActiveX, CORBA, or JavaBeans protocols. Other communication protocols or component architectures are or will become available and perhaps widely used, and it should be understood that they are also applicable to the present invention.

Figure 12 is a flow chart illustrating the process diagrammed in Figure 11. At 1550, the client computer first sends to the web server 1504 a request for a particular document, which is then serviced by the web server 1504 by sending the requested document back to the client 1502. The browser program 1520 informs the program 1525 of the identify of the document that was just requested and displayed (1554). The program 1525 then sends information to the server 1506, inquiring whether supplemental information is available for the displayed document (1556). The server 1506 responds by either informing the program 1525 of the location of supplemental information available for the identified document or by informing the program 1525 that there is no supplemental information available for the displayed document (1558). If supplemental information is available for the displayed document, the program 1525 directs the browser to retrieve the supplemental information (e.g., from server 1508) and display it on the screen with the displayed document (1562, 1564, 1568, and 1570).

Figure 13 is a network diagram of another embodiment of the present invention, and Figure 14 is a flow chart illustrating the process diagrammed in Figure 13. As shown in Figure 13, the browser program 1530 executing on the client computer 1502 first retrieves from the web server 1504 a given document or resource, and the browser 1530 informs the program 1535 of the identity of the retrieved document (1572, 1574, and 1576 in Figure 14). The program 1535 then sends the identity of the retrieved document to the server 1506, and in response, receives the location of where any supplemental information for the retrieved document can be found (1578 and 1580 in Figure 14). The program 1535 then retrieves the supplemental information and presents it to the user, typically by displaying it on the screen. The embodiment of Figure 13 thus differs from

that of Figure 11. One difference is that unlike program 1525, program 1535 of Figure 13 displays the supplemental information rather than directing the browser to display that content.

In Figures 11 to 14, the programs 1525 and 1535 may be informed of the identity of the document presented to the user by the programs 1520 and 1530 through interprocess communication. It may be appropriate in other embodiments, however, for the program 1525 (or the program 1535) to access the network connection or network stack or operating system software directly, and analyze the information or packets being passed over the network. The data on the network may need to be analyzed and/or decoded to decipher the necessary information about the identity of the document presented to the user. It should be understood that any known or hereafter developed technique for identifying the content being presented to the user could be used, and such techniques may also be applicable to the present invention. The present invention need not be limited to the interprocess communication techniques described in connection with Figures 11 to 14.

Figure 15 is a flow chart that illustrates the operation of a server computer such as computer 1506 in the embodiments diagrammed in Figures 11 and 13. In Figures 11 and 13, a server process executes on the computer 1506 to provide information to client computers about where supplemental information relating to network resources can be found. At 1710 and 1712 of Figure 15, the server process waits for a request from a client computer. Upon receiving a request, the server process looks up the document specified in the client's request in a look-up table or some other data structure (1714).

If the server does find at 1714 and 1716 that there is supplemental information for the specified document, the client computer is so informed at 1718, and the client computer is also informed at 1720 where the supplemental information can be found. Where there is no supplemental information for the specified document, the reference server informs the client that no entry exists for the specified document, which the client understands to mean that there is no supplemental information for the specified document. The loop 1710 and 1712 is then reentered.

Figure 16 is a simplified representation of a database or look-up table such as that referred to at 1714 in Figure 15. As can be seen from this table, the "Flying Over the Precipice" document that is stored on the "Universal Theatres" server has an entry at 1751 specifying that supplemental information for this document can be found at the server called "reference" and within the

document called “flying over the precipice.” This is the document displayed in window 320 in Figure 4. There are also other entries 1753 for the “Flying Over the Precipice” document in the table of Figure 16, which means that there are a few instances of supplemental information for the “Flying Over the Precipice” document.

There are entries for other documents in Figure 16, such as the “index” document from the “Hunan Taste” server. This document has supplemental information available, and that supplemental information is contained within the document “metro pages quote” and is stored on the “oriental specialties” server. Similarly, the “dine-in menu” document on the “Hunan taste” menu also has supplemental information within document “dine-in menu” at the “oriental specialties” server. By referring to the entries in the look-up table, the server is able to determine whether supplemental information is available for any given document. In the embodiment shown in Figure 16, if there is no entry in the look up table for a document, no supplemental information is available for that document.

In some embodiments of the present invention, each time a new document or resource is presented to the user (e.g., in window 306 of Figure 3), a request for supplemental information is always initiated. In other embodiments, the client computer only occasionally sends a request over the network to determine whether there is supplemental information for the document displayed or otherwise presented at the client computer. Figure 17 is a flow chart illustrating a process in which the system waits at 1302 until the user requests that a new document be displayed. When the user makes such a request, the requested document is displayed at 1306. If the requested document is not from the same server as the previous document, supplemental information relating to a previous document is removed from the display at 1309. A message is then sent over the network to a reference server so that the reference server can determine whether supplemental information is available for the displayed document (1310, 1312). If so, the supplemental information is retrieved and displayed. However, if the displayed document is from the same server as the previous document, or if there is no supplemental information for the displayed document, the waiting loop 1302 is entered.

The embodiment of Figure 17 thus operates to check the reference server for supplemental information only where the user requests a document that is stored on a server that is different from that of the previously-displayed document. The amount of network traffic generated by repeated

requests to the reference server is reduced in this embodiment because the requests are generated only when the user accesses to a new server. In a modified version of this embodiment, the reference server is checked only where the domain has changed.

Figure 18 is a flow chart illustrating a process in another embodiment of the present invention. In Figure 18, if it is determined at 1408 that the requested document is not stored on the same server as the previous document, then a reference server is checked for supplemental information for the displayed document (1410). If supplemental information is available, that information is retrieved and displayed (1412). In addition, the reference server is checked for supplemental information for other documents stored on the document server, and this information is stored on the client computer (1416). The loop 1402 is then entered. If at 1408 the requested document is from the same server as the previous document, the locally stored information for other documents stored on the same server is analyzed. This information may have been previously stored at 1416, and if it reveals that supplemental information is available for the displayed document, the supplemental information is retrieved and displayed at 1428.

Thus, in the embodiment of Figure 18, that portion of the look-up table or database (or whatever data structure is used) that corresponds to a particular server is downloaded to the client the first time that the client computer retrieves a resource from a particular server. Thereafter, the client need not use the network to determine whether supplemental information is available for any other documents stored on that same server because the information necessary to make this determination is stored locally on the client. This embodiment may be more efficient where the cost of making multiple requests over the network is high and the information that needs to be downloaded each time a new server is encountered is not too voluminous.

In Figure 19, a window 506 from a document retrieval program or a browser is shown on the screen 504. The title bar 507 identifies the server from which the document was retrieved (i.e., the server controlled by “Hunan Taste, Inc.”), and the identity of the document (i.e., “index”). This “index” document relates to a restaurant named “Hunan Taste,” located in a hypothetical city’s “Redwood District.” This document provides links to information about the Hunan Taste restaurant’s daily specials and dine-in and carry-out menus, which can be retrieved by the user by selecting one of the display elements 532. A delivery order from Hunan Taste can also be placed by selecting display element 534. A banner advertisement 512 promotes an establishment called

“Jerico’s Tavern,” which is located near Hunan Taste. In this embodiment, Jerico’s Tavern advertises on Hunan Taste’s page because it seeks to attract patrons leaving Hunan Taste after dining. Because Jerico’s Tavern is not a direct competitor to Hunan Taste, Hunan Taste may permit (even encourage) Jerico’s Tavern to advertise on the page shown in window 506.

When the “index” document from the Hunan Taste server is retrieved over the network and displayed in the window 506, as shown in Figure 19, the window 520 also appears on the display screen 504. The information shown in window 520 relates to a competing Chinese restaurant called “Oriental Specialties.” A person viewing the Hunan Taste page shown in window 506 is likely to be interested in going to a Chinese restaurant, so “Oriental Specialties” has an interest in providing an advertisement to such a person. In some embodiments, however, Hunan Taste may not allow a competing restaurant (i.e., Oriental Specialties) to advertise on the page displayed in window 506. Thus, Oriental Specialties may be unable to reach these potential customers through advertisement on Hunan Taste’s pages. (In other embodiments, Hunan Taste may permit such activity.)

The window 520 allows Oriental Specialties to communicate with the person viewing the Hunan Taste web page displayed in window 506. These persons may be considering a visit to the Hunan Taste restaurant, or at least may be interested in Chinese food. In window 520, Oriental Specialties provides information targeted specifically to individuals viewing the Hunan Taste page shown in window 506 by quoting a magazine article in which Oriental Specialties is favorably compared to Hunan Taste. This may be targeted and effective marketing for Oriental Specialties because the message displayed in the window 520 is likely to be seen by individuals looking for the very service that Oriental Specialties provides. In addition, the person viewing the Hunan Taste page may also be interested in seeing the information shown in window 520. Because this person has retrieved the Hunan Taste page, he or she likely is interested in Chinese cuisine, yet may not know about the Oriental Specialties restaurant. Thus, from the user’s perspective, the window 520 may provide useful, relevant, and appreciated information that might not otherwise be seen when viewing the Hunan Taste page alone.

In Figure 20, the “index” document from the Hunan Taste, Inc. server is displayed in window 606, and in this embodiment, a task bar 608 is also shown on the display screen 604. This task bar 608 has buttons 612 that can be used to select particular programs or stores of

information. In Figure 20, supplemental information is displayed in response to the user selecting (e.g., through an input device) the display element 610.

In Figure 21, the user has selected the display element 610, causing supplemental information similar to that shown in window 520 in Figure 19 to be displayed in the box 607. Although the button 610 is provided on the task bar of Figures 20 and 21, it need not be, and such a button or other functionality could be provided, for example, through a menu structure, or through an icon elsewhere on the screen.

In another embodiment of the present invention, shown in Figure 22, supplemental information relevant to the information displayed in a browser window 806 is displayed in a scrolling window 810 shown on the taskbar 808. The supplemental information may be scrolled in window 810 where the window 810 is quite small.

Figure 23 shows a window 1806 on a display screen 1804. As indicated in the title bar 1808 for window 1806, the document “widget purchase page” from the XYZ Corporation server is displayed in window 1806. This page may be used to purchase a specified quantity of “widgets,” which are products manufactured and sold by XYZ Corporation. A person browsing this page can purchase a quantity of XYZ Corporation Widgets by entering a credit card number (or other financial account number) and specifying the manner in which the Widget(s) should be shipped. (This information could also be provided automatically.) Once the requisite information is completed by the user, the user selects button 1820 to place the order and to process the transaction. Window 1819 is shown in Figure 23 to illustrate how the display screen 1804 may appear when supplemental information is available for a particular page but, because of limitations in network resources or bandwidth, has not yet been fully retrieved over the network and rendered on the display. The window 1819 in the embodiment of Figure 23 informs the user that there is supplemental information for the XYZ page shown in Figure 23, and acts as a “place holder” while the supplemental information is being retrieved over the network or rendered on the display.

Once the supplemental information is retrieved over the network, the display of Figure 23 is updated to that shown in Figure 24. Window 1820 now includes an advertisement from ABC International, Inc., in which that entity offers to sell widgets for a price that is lower than those sold by XYZ Corporation. Thus, as shown in Figure 24, despite the fact that ABC International, Ltd. is a direct competitor to XYZ Corporation, ABC is effectively able to advertise in a manner closely

associated with XYZ Corporation's pages, and even at XYZ's point of sale. This can be a very effective scenario for advertisement, precisely because the advertisement is so pertinent and of interest to the user viewing this particular XYZ Corporation page.

If the user selects the advertisement in window 1820, the display is, in one embodiment, updated to that shown in Figure 25. Figure 25 is a point of sale page for ABC International, where the user can specify the quantity of widgets that he or she wishes to purchase, and can purchase them at this page. Thus, the user may be able to move very efficiently from the point of sale at the XYZ Corporation page to the point of sale page at a competing manufacturer, and compare prices and/or features and selection very efficiently, quickly, and easily. In another embodiment, a page for ABC International is displayed in a separate window or other display area, leaving the XYZ page at least partially visible on the display.

As illustrated in the look-up table of Figure 16, some network resources may have more than one supplemental information document or resource that is relevant to it. For example, as shown in Figure 16, the widget purchase page on XYZ Corporation's server has three different entries in the look-up table in Figure 16. One way in which this situation is handled is to simply choose a single entry for a given document or resource and provide only that supplemental information document. Selection of the entry to be used can be done randomly, or through a weighted (or evenly-weighted) average for each of the available entries. Another method is to display a large number or all of the entries.

In another method of handling such a situation, illustrated in Figure 26a, the three supplemental information documents are rotated in the window 1820 in Figure 24. Each advertisement may be paused for a preprogrammed period of time, which may differ for each advertisement. Figure 26a shows the order in which each of the three advertisements are displayed in the window. This method allows multiple third parties to provide supplemental information that is relevant to single page. The amount of time that the advertisement or other message is displayed could depend on the rate paid by the advertiser. It could also depend on the number of advertisers or information providers that wish to provide information in this manner. In the embodiment of Figure 26a, when the time period for the third supplemental information document (i.e., JSO Technology) is completed, the order repeats.

When supplemental information is presented in the manner illustrated in Figure 26a, where each is allocated a portion of time on the display, the user may have difficulty selecting supplemental information when the time allocated for one or more instances of supplemental information is short. For example, if each of the three instances of supplemental information illustrated in Figure 26a is displayed on the screen for only three seconds, the user may have to act quickly in order to move the pointing device to the window 1820 (or perform some other method of selecting the information in the window 1820) in Figure 24 when the desired supplemental information is displayed or otherwise presented to the user. If the user waits too long, the next instance of supplemental information will be rotated into the window and the prior instance will be lost (at least temporarily). This problem can be addressed by suspending the timer (or some other similar mechanism) that keeps track of the supplemental information during the time that the user is moving the cursor. For example, while the user is substantially continuously moving the cursor 1801 (or otherwise activating the input device) on the screen in Figure 24, the supplemental information displayed in window 1820 will suspend or delay the process of rotating the next instance of supplemental information into the window 1820. Thus, if the user starts moving the cursor 1801 on the display just prior to the time for the instance of supplemental information displayed in the window 1820 to be changed, the supplemental information may remain until the user stops moving the cursor. This provides a more forgiving environment for selecting transient display elements such as the window 1820.

Figure 26b is a flow chart illustrating an implementation of such a procedure. At 2650, supplemental information is presented to the user (e.g., displaying information on the display screen, playing an audio clip). At 2652, a determination as to whether the supplemental information that is currently being presented to the user should be replaced with other supplemental information is made (e.g., determining if a predetermined amount of time has elapsed, or if some other condition is satisfied). If not, the process of Figure 26b checks to see if the input device is activated at 2654. If the device is not activated, the condition at 2652 is again tested. This loop continues as long as the user's input device (e.g., pointing device, keyboard, voice recognition logic) is not activated, and the condition at 2652 is not satisfied. When the input device is longer activated, a loop at 2654 continues until that input device is no longer active. And when a determination is made that the supplemental information should be replaced, new supplemental

information is selected (e.g., as described in connection with Figure 26a) at 2656, and then presented to the user at 2650.

In Figure 27, the window 2006 includes within it an area 2009 in which supplemental information is displayed. This area 2009 is integrated into and may be part of the window 2006. A button 2011 may be used to hide and unhide the area 2009. In the embodiment shown, selection of the button 2011 when the supplemental information area 2009 is displayed will remove from the window from the area 2009. Thereafter, the button 2011 will still be visible, and selection of that button will restore the area 2011. In other embodiments, the area 2009 may be positioned in many different areas and in many different ways. For example, the area 2009 could be configurable within the browser, and retrievable through a keyboard command or menu selection.

A flow chart of a process for presenting a supplemental resource or supplemental content to a user is illustrated in Figure 28. The system waits at 1202 until the user requests that a network resource be presented. At 1204, the presentation of any supplemental resource(s) relating to an earlier presented resource may be discontinued. (In another embodiment, the presentation of such a resource may be continued.) The requested resource is then retrieved and presented to the user at 1206. The presentation of this resource may involve an animation, the execution of a program, the playing of a video or audio clip, the display of a text-based document, or any combination of these or other techniques. At 1208, the system determines whether there are supplemental resource(s) available for the currently presented resource. As described previously, in one embodiment, this may involve a request sent to a reference server that is different from that from which the currently presented resource has been retrieved. Alternate procedures could also be followed. If supplemental resources are available at 1210, at least one of these resources is retrieved at 1211 and presented to the user at 1212. These resources may be a computer program, a video or audio clip (or both) or any other form or type of content or multimedia.

In Figure 29, a browser window 2901 is shown on a screen, and a page from the "Hunan Taste, Inc." server is rendered in the browser window 2901. On the task bar 2907 is an icon, button, or display element 2905. In Figure 30, the display element 2905 of Figure 29 has been selected with the cursor 2903, and the appearance of the display element 2903 has changed. In the embodiment shown, the display element 2903 has changed shape and is shaded, but any other visual or other indication technique may be used. After a period of time, the display is further updated to

that of Figure 31, where a window or area 3201 is displayed on the task bar 2907. Within the area 3201, various display elements 3203, 3205, 3207, and 3209 are displayed.

Each of the display elements 3203, 3205, 3207, and 3209 represent a type of supplemental information relating to the content or information displayed in window 2901. In the embodiment shown, the presence of four display elements 3203, 3205, 3207, and 3209 means that four different types of supplemental information are available for the content rendered in the window 2901. For an embodiment like that of Figure 31, if only one type of supplemental information were available for the content rendered in window 2901, only one display element would be displayed in the box 3201. In another embodiment, however, a display element may be present for each type of supplemental information, whether or not that type is available. In such an embodiment, display elements may be modified or highlighted when supplemental information of the corresponding type is actually available.

In Figure 32, the display has been updated following the selection of display element 3203 with the cursor 2903. The selection of the display element 3203 causes supplemental information of the type represented by display element 3203 to be displayed in the window 3303 in Figure 32. The display element 3203 represents supplemental information relating to competing services, or services that compete in some way with a given service provider. Because the service provider that is the subject of the content rendered in the window 2901 in Figure 32 is the Hunan Taste restaurant, competing services may be, for example, other restaurants. And as shown in Figure 32, window 3303 displays supplemental information about a competing restaurant, Oriental Specialties. A display element 3305 is shown in the window 3303. This display element 3305 indicates which type(s) of supplemental information are being provided in the window 3303.

Figure 33 is a network diagram illustrating the network activity underlying Figures 29 to 32. At computer 902, the user operates the browser or document viewing program represented by the window 2901 in Figure 29. The computer 902 sends a request for a document to the computer 904, which is the Hunan Taste, Inc. server. In response, requested document is sent by the computer 904 to the computer 902. See the arrows labeled "1" in Figure 30. At this point, the display is as shown in Figure 29.

When the display element 2905 is selected in Figure 29, the display is updated to that shown in Figure 30, and a request is sent by the computer 902 to computer 906 for information about the

types of supplemental information available for the content rendered in window 2901. The display is updated to that shown in Figure 30 to indicate to the user that information describing the types of supplemental information available for the content of window 2901 is being retrieved over the network. Because of network or other limitations, the retrieval of this information may take time. Computer 906 responds to the request made by computer 902 by sending information about the types of supplemental information (if any) available for the content rendered in window 2901. See the arrows labeled “2” in Figure 30. When the supplemental information types are retrieved, they are communicated to the user, such as in the manner shown in Figure 32. In some embodiments, the computer 906 may also send the location and identity of supplemental information for one or more information types, thereby making this information available for later use.

When the user selects the supplemental information type in Figure 32, supplemental information of the selected type is retrieved over the network (e.g., from computer 908). See the arrows labeled “3” in Figure 30. The location of this supplemental information may have been identified when the types of supplemental information were previously retrieved from the computer 906. (See arrow “2” of Figure 30.) If not, however, the location of the supplemental information may be retrieved over the network from computer 906 (or from another computer on the network) before the supplemental information is actually retrieved from computer 908.

Figure 34 shows numerous display elements, each of which may represent a different type of supplemental information in the window 3201 of Figures 31 and 32. Display element 3415 was the display element selected in Figure 33, and it represents supplemental information for competing service providers. For a page or site relating to a restaurant, supplemental information from a competing service provider may be information about another restaurant. For a service provider such as, for example, a landscaping service, supplemental information from a competing service provider may be information or content from another landscaping service, or perhaps a lawn maintenance service. Since there are service providers for almost every type of service, a display element of this nature can be used in a wide variety of other contexts, and in many situations.

Display element 3413 represents supplemental information or content relating to a competing product supplier. Supplemental information of this type for a page involving bicycles, for example, may involve information about another bicycle manufacturer or another bicycle distributor. This display element may represent supplemental information relating to any type of

product. A display element of this nature can be used in a wide variety of other contexts, and in many situations.

Display element 3411 represents supplemental information or content from a competing information supplier. This display element may be pertinent to a page from a news or business information provider. Selection of display element 3411 may cause information or content to be presented to the user that is published by a competitor of the publisher of the information already being presented to the user. For example, for a page from a sports information site that publishes the latest sports scores and highlights, supplemental information of this type may be information about a competing sports information site. A display element of this nature can be used in a wide variety of other contexts, and in many situations.

As described, the display elements 3415, 3413, and 3411 described above present information from or about a competitor to the subject of the information or content that is otherwise being presented to the user. The display element 3415 represents competing service providers, and may be useful at least when the user is viewing or is being presented with content relating to a particular service provider. The display element 3413 represents competing product information, and may be useful at least when the user is viewing or is being presented with a page describing a particular product. And the display element 3511 represents competing information, and is useful at least when the user is being presented with content or information of a particular type (e.g., news or business information).

Display element 3409 represents supplemental information relating to price, and may be used to indicate that price information is available for the content or page being viewed by the user. For a page offering a particular product for sale, supplemental information of this type may describe a competitor that offers the same or a similar product at a reduced price. Alternatively, for the same page, supplemental information of this type may be information from an entity that provides cost analysis or cost commentary for the offered product or for the company offering the product. Another example of the use of this type of supplemental information is shown in Figure 24. A display element of this nature can be used in a wide variety of other contexts, and in many situations.

Display element 3407 represents a type of supplemental information that relates to the actual information or content being presented to the user. For example, for a movie or video,

display element 3407 may represent a review of that movie or video. For a news page, display element 3407 may represent a review or critique of the publisher of the news page, or of the particular news page itself. This type of supplemental information may differ from that of, for example, display element 3411 where it provides information about the content itself, rather than similar, competing content. A display element of this nature can be used in a wide variety of other contexts, and in many situations.

Display element 3405 represents supplemental information of a type that is information about a given product, service or instance of information. For a given product or service, supplemental information of the type represented by display element 3405 may be a review or critique of that product or service. See, e.g., Figure 4. In another example, for a product that requires assembly, supplemental information of this type could include instructions on how to assemble the product. A display element of this nature can be used in a wide variety of other contexts, and in many situations.

Display element 3403 represents supplemental information about a service. For example, for a page describing or offering for sale a vacuum cleaner, the display element 3403 may represent information about a home or office cleaning service. Or for a page describing how to select winners in the stock market, display element 3403 may represent information from a discount brokerage house or a electronic stock trading service. A display element of this nature can be used in a wide variety of other contexts, and in many situations.

Display element 3401 represents supplemental information about a product. For example, for a page or video that describes how to construct landscape retaining walls, display element 3401 may represent information from the manufacturer or distributor of a tool that easily splits the retaining wall blocks, as may be required when constructing such a retaining wall. A display element of this nature can be used in a wide variety of other contexts, and in many situations.

The foregoing display elements are described merely to illustrate the various types of supplemental information that may be made available, and these categories are not meant to be exhaustive. And they may not be rigidly defined in every embodiment. Further, different schemes for categorizing supplemental information can be used, and it should be understood that any such alternate schemes for categorizing supplemental information may be applicable to the present invention.

Figure 35 is a flow chart of a process similar to that illustrated in Figures 29 to 32. At 3501, the user retrieves a network resource over the network. The network resource is then presented to the user at 3503. This may involve the content being displayed for the user, or it may involve an audio presentation or any other type of multimedia presentation. The user then requests at 3505 information about the types of supplemental information. This request can be made by selecting a button with a pointing device (e.g., display element 2905 in Figure 31), or this request could be made through any other input device (e.g., voice recognition). The request at 3505 causes information about the types of supplemental information available to be retrieved over the network at 3507, and communicated to the user at 3509. In one embodiment, the types of supplemental information available are communicated to the user through display elements as are shown in Figures 31 and 32.

Still referring to Figure 35, the system loops until the user selects a type of supplemental information at 3511. When such a selection is made, supplemental information of the selected type is presented to the user at 3513. This supplemental information may be retrieved over the network, or, if it was already retrieved over the network, it may be retrieved from local storage.

Figure 36 is a flow chart illustrating an alternate process. At 3601, the user retrieves a network resource over the network. The network resource is then presented to the user at 3603. At 3605, information describing the available types of supplemental information is retrieved over the network, without waiting for the user to request this information. The user is informed of the types of supplemental information available at 3607, and supplemental information is presented to the user at 3611 when the user selects a type of supplemental information at 3609.

Figures 38a and 38b illustrate display screens in an embodiment in which rendered in the browser window is supplemental information that relates to an earlier page rendered in the browser window. In other words, in one embodiment, supplemental information relating to the previous page is provided when the user accesses another a site on the network that acts as a server for supplemental information. In Figure 38a, the browser window 3801 shows a page from the Hunan Taste, Inc. server. Supplemental information is provided to the user about the Hunan Taste page shown in window 3801 when the user communicates to the browser program the identity and/or location of a server for supplemental information. (The identity and/or location of such a server could be communicated to the browser a number of different ways, such as by typing it into a text

box within the browser program, selecting such a server from a pull-down menu, selecting an display element or link within the browser window that corresponds to the supplemental information server, or any other way known in the art.)

When the identity and/or location of such a server is provided to the browser, the browser communicates to that server the page or resource that is currently being presented to the user (i.e., the Hunan Taste page). The server for supplemental information evaluates this request, and then serves supplemental information. This supplemental information is then presented in the window 3801 as shown in Figure 38b, replacing the content in window 3801 in Figure 38a.

Figure 37 is a flow chart of a process similar to that described in connection with Figures 38a and 38b. At 3701, the user of a browser program (or a similar program) retrieves a network resource, which is termed network resource “alpha” in Figure 37. This network resource alpha is retrieved from a server (or servers) on the network. The resource alpha is then presented to the user at 3703. Thereafter, the user goes to a supplemental information site on the network (3705), and the identity of the network resource that was previously presented to the user at 3703 (i.e., “network resource alpha”) is communicated to the supplemental information site (3707). Based on this information, the supplemental information site provides supplemental information at 3709 about the previously-presented network resource (i.e., network resource alpha). This supplemental information is presented to the user at 3711.

Figure 39 illustrates a simplified display screen in another embodiment of the present invention. In Figure 39, the display element 3905 in window 3901 has been selected with the cursor 3905, causing window 3907 to be displayed. The window 3907 provides supplemental information of the “competing service” type. Competing service information from three different competitors is shown in window 3907 (Oriental Specialties, Thai Taste, and The King’s Kitchen). Information from each of the three different competitors is displayed in its own frame or sub-window 3909, 3911, and 3913. An alternate implementation may have three or more windows, one for supplemental information from each of the competing restaurants (or other entities).

Figure 40 is a diagram of a network where supplemental information from more than one source is presented to the user. Initially, the user at computer 902 requests and then retrieves a network resource from server 904. This request and response from the server 904 is indicated by line 4001. The identity of the resource retrieved from computer 904 is then communicated by

computer 902 to server 906, which responds by identifying the location of supplemental information about the resource retrieved from server 904. This communication and response is represented in Figure 40 by line 4002. In the embodiment shown, the identifying information received by computer 902 from server 906 points to servers 920, 922, and 924. Thus, supplemental information is retrieved by computer 902 from each of the servers 920, 922, and 924, as is indicated by the lines 4003, 4004, and 4005. In this embodiment, supplemental information is distributed throughout the network, and it is retrieved by the client computer from a number of different servers.

In an embodiment that differs from that shown in Figure 40, supplemental information from or relating to more than one source could be stored on a single server. This would allow some, perhaps all, of the supplemental information to be retrieved from the same server.

In Figure 41, window 4101 displays information relating to the hypothetical Acme Electronics Corporation. Acme Electronics may manufacture or sell a large variety of products, and only a general page from Acme Electronics is displayed in window 4101. Window 4102 displays a number of display elements representing different types of supplemental information available for the Acme Electronics server. When cursor 4104 selects display element 4103, which represents supplemental information relating to competing products, another window 4105 appears. The window 4105 displays six different categories of products that the user can select. These categories represent products or categories of products that are manufactured and/or sold by Acme Electronics Corporation.

When the user selects one of the products (or categories of products) in Figure 41, supplemental information that is from the selected category is presented to the user. In Figure 42, the user has selected the “widgets” category in window 4105 by selecting the “widgets” text in the window 4105. This causes the window 4110 to be displayed, in which supplemental information for three widget manufacturers or distributors that compete with Acme Electronics Corporation is presented.

Because of the nature of some types of networks, it may be difficult to determine which document from a server is being retrieved by a client computer or presented to a user at a client computer. In other words, for embodiments such as those described in connection with Figures 11 to 14, even where it is possible to determine which computer or server is being accessed on the

network (through the use of a domain name or IP address), it may be difficult to accurately determine which resource or resources on that computer are being accessed. For example, where documents are generated dynamically by the server, they may not have a consistent name or URL. Also, encryption techniques or the like can be used to make determination of the nature of the document being presented to the user difficult. Similarly, documents could be continually modified, at least slightly, to make consistent identification and differentiation from other documents difficult. In addition, program 1520 or 1530 may not be capable of providing the identity of the document being retrieved or presented. Thus, particularly where the person or entity in control of a server wishes to make accurate identification of a resource on the server difficult, the identity of the specific resource or document that is being presented to the user may be difficult to determine. A menu such as that shown in Figures 41 and 42 may be useful in such a situation because it allows the user to select which type of information he or she is seeking at a particular site. Thus, one at least partial solution is to provide a menu so that the user can select the type of supplemental information that he or she wishes to see.

Figure 43 shows an embodiment similar to that of Figures 41 and 42, but where a limited number of categories of products are displayed in a scrolling window 4105. In the window 4105 in Figure 43, the categories are scrolled through the window so that only a limited number of categories are displayed at a time. The marker 4108 marks the position in the list of categories. When the end of the list is approached, this marker 4108 moves toward the bottom of the slot 4115. When the end of the list rolls over to the front of the list, the marker 4108 moves to the top of the slot 4115. The visual indicator 4109 marks where the list begins repeating. This embodiment takes a small amount of screen real estate, yet allows the user to select from a large list of categories without searching for the desired category. In a manner similar to that described in connection with Figures 26a and 26b, the categories in the window 4105 may, in some embodiments, discontinue scrolling when the cursor is being continually moved on the display.

The menus of Figures 41 to 43 are shown in connection with various implementations of the present invention, but it should be understood that other types of menus, perhaps having additional levels, could be used where appropriate. Further, the menus shown in Figures 41 to 43 are merely illustrative, and it should be understood that many other menu structures or user interfaces known in the art or hereinafter developed may be applicable to the present invention.

In Figure 44, the window 4405 displays information relating to the Acme Electronics Corporation in a manner similar to that shown in Figure 41. A display element or button 4401 is shown on the task bar 4409. The user selects button 4401 when he or she wishes to see categories of supplemental information. In Figure 44, the user has selected display element 4401, thereby causing window 4403 to be displayed as shown in Figure 44. This window 4403 communicates categories of products and services available from Acme Electronics Corporation. The window 4403 also indicates the types of supplemental information available for each of the categories. For example, for the “printers” category, supplemental information about services relating to printers manufactured or sold by Acme Electronics is available. Price information about the printers manufactured or sold by Acme is also available. In addition, information about printers manufactured or sold by competitors to Acme is available.

In Figure 45, the user is selecting with the cursor 4511 the display element 4501, which represents a type of supplemental information in the “widgets” category of window 4403. The display element 4501 that is being selected by the user is the competing product icon for the widget category. After the display element 4501 is selected, the display is updated to that shown in Figure 46, where supplemental information relating to competing suppliers of widgets is shown at 4601. In the embodiment of Figure 46, the categories and supplemental information type display elements remain on the screen while the supplemental information is presented to the user. In another embodiment, shown in Figure 47, the competing product supplemental information is presented in the window 4701, but the categories and supplemental information type display elements, shown in window 4403 in Figure 45, are removed from the screen.

Referring again to Figure 45, one or more display elements are shown for each of the categories shown in window 4403. Although Figure 45 has been described in connection with the selection of display element 4501, the user may select other display elements (e.g., 4502, 4503) for the widget category. Where other display elements are selected, supplemental information that corresponds to the selected display element would be presented to the user. Display elements from any of the other categories shown in window 4403 could also be selected, which will result in the presentation of appropriate supplemental information for the selected category and supplemental information type.

The system of Figure 45 could also be configured to allow the selection of the category itself, without selection of any particular display element for the category. When the user selects the category (e.g., by selection of the text describing the category), supplemental information of all types may be presented to the user. Such an embodiment may be useful where the user wishes to see all of the supplemental information for a particular category.

In Figure 48, an application that presents audio content to the user is represented by window 4805. The audio content is published by an entity called “WCAR,” and the audio content is a play-by-play broadcast of a “Cardinals” football game. As shown in the window 4805, the Cardinals lead that game 21-20 in the second quarter. In window 4807, a full motion video from the Universal Theatres server being presented. The video shown in window 4807 is the “Flying Over The Precipice” movie. Thus, for the screen of Figure 48, audio content is presented to the user in window 4805, and video content is presented to the user in window 4807. A portion 4809 of the window 4805 is shaded or colored or otherwise marked so as to communicate to the user which of the window on the display is the “active” window. Because the content relating to both of the windows 4805 and 4807 present audio to the user, it may be desirable to present audio from only one source. In Figure 48, the window 4805 is shown to be the active window, and it may have been placed in that condition by selection by the user (e.g., using the cursor 4803), by default (e.g., it may be the most recent window to appear on the screen), or in any other way. In the embodiment of Figure 48, audio from the active window 4805 is presented to the user, while the audio from window 4807 is muted. The motion video in window 4807 may or may not continue.

Still referring to Figure 48, the user selects the display element 4803 in the box 4801, which causes the display screen to be updated to that shown in Figure 49. In Figure 49, supplemental information about other competing content is provided in the window 4901. The supplemental information displayed in window 4901 describes a live audio broadcast of a Gophers football game, where the Gophers are leading 14 to 3 with 12:19 left. This information is related to the content being presented to the user in window 4805, the active window in Figures 48 and 49. The window 4901 also indicates that a commercial being broadcast in connection with the Gophers game will end in only four seconds. This type of information may encourage the user to select the content described in the window 4901. Similar types of information may be provided in other situations. For example, if the Gophers game were at half-time, the user may switch to a broadcast of another

game (e.g., the Cardinals game) while the Gophers game is at half-time. The window 4901 in such a situation may allow the user to keep track of when the Gophers game resumes for the second half. Similarly, the user can keep track of when a particular program or sporting event (broadcast via audio, video, or otherwise) ends, or when the Gophers game gets closer than it currently is. All this may be done while listening to the Cardinals game, or when other related content is being presented to the user.

When the user selects the window 4901 in Figure 49, the broadcast of the Gopher game is presented to the user, and the broadcast of the Cardinals game is turned off, or it may be reduced in volume. In Figure 50, the user has selected the content represented by the window 4901 in Figure 49, thereby causing the display of Figure 49 to be updated to that of Figure 50. Window 4901 is the active window in Figure 50. The window 4901 of Figure 50 shows the score of the Gophers game. The window 4805 may or may not remain on the screen. If it remains on the screen, it may be continually updated so that the score of that game is presented to the user, at least visually. In another embodiment, supplemental information about the content being presented in connection with the active window 4901 is presented to the user. This supplemental information may relate to the Cardinals football game, or it may describe another type of supplemental information.

Supplemental information relating to the video presented in window 4807 in Figures 48 and 49 could also be presented to the user (e.g., in window 4901). For example, supplemental information could describe content that starts (or that can be configured to start) just as the video rendered in window 4807 completes. Alternatively, supplemental information from the manufacturer of a product that is used or shown in the movie (e.g., an automobile or clothing) could also be presented. Thus, supplemental information can be presented in connection with many types of content, and need not be limited to text, but can fully encompass audio, video, animation, and any other type of content.

For purposes of illustration and example, the network documents described and illustrated in many of the Figures herein are shown displayed within a window controlled by an application program, such as may be represented by window 306 in Figure 3. It should be understood, however, that such documents (or resources) could be displayed in other ways. For example, the contents could be displayed on the entire desktop, or a portion of the desktop. In another embodiment, the contents might be scrolled on the screen, perhaps under other windows. Further,

windows could be controlled by the operating system, rather than by an application program. The manner in which information is being presented to the user herein is for illustration purposes only, and should not be interpreted as limiting the present invention.

Further, the present invention has been described principally in terms of computer network-available documents containing text, graphics, animated graphics or other objects. It should be understood that the present invention is not limited to only those particular types of network resources. Rather, the present invention can be implemented using audio, video, or other types of multimedia beyond that shown. For example, the supplemental information could be an audio clip, or an applet or application program that carries out a useful function that relates to the resource displayed or otherwise presented to the user. In addition, the term "computer" as used herein should be interpreted broadly to encompass all systems, devices, or machines that have a general or specialized processor or microprocessor or a similar device known in the art or hereinafter developed.

Also, embodiments of the present invention may be integrated with one or more payment or transaction processing systems or the like so that a person viewing certain types of content and/or certain types of supplemental information or supplemental content will incur a fee or a transaction cost. Authentication and automatic payments techniques may also be used.

In a networked environment, such as that of Figure 2, there are numerous ways in which software can be installed, distributed, and/or executed on the various computers on the network. Figure 51 illustrates a conventional way in which desktop software is installed and executed. In Figure 51, a computer program 1003 is installed at the computer 1001 through some type of installation program typically started by the user of the computer 1001, and executed on the computer 1001. During installation, the program 1003 may need to be configured at the computer 1001 for use with the network in order to enable access to other computers on the network (e.g., 1002a and 1012). After installation, the computer program 1003 resides and executes at the computer 1001, and is persistent. When the computer 1001 is shut down or restarted, the program continues to be stored at the client on non-volatile storage media. Upon restarting the computer 1001, the program 1003 is available for use without reinstallation.

Figure 52 shows a different embodiment. When the network-connected computer 1001 connects to or downloads an object stored on the remote computer 1002a over the network, a

program 1005 embedded within the downloaded document or object is installed on the computer 1001 and is executed on the computer 1001. Figure 53 is a flow chart that illustrates one possible installation procedure that is carried out when the computer 1001 accesses the program 1005.

The computer 1001 identifies at 1020 one or more programs embedded within the accessed object. The client computer then determines whether each embedded program has been installed previously on the computer 1001. This can be done by searching the computer's storage or system registry for the program or for the program's identifying characteristics. In Microsoft's ActiveX/COM architecture, for example, this is done by searching the registry for an instance of the program's globally unique identifier (GUID) in the system registry.

If the embedded program has been installed on the client computer, the previously installed program is retrieved from local storage at 1030, and executed at 1028. However, if the program has not been already installed on the client computer, it is retrieved over the network (1023), and installed on the client computer. The installation process will typically involve updating a system registry or other persistent storage with identifying information on the computer 1001.

Preferably, the program is installed at 1024 such that it need not be downloaded again over the network when it is encountered embedded within another object. For example, if the computer 1001 were to access an object on computer 1012 that had program 1005 embedded within it, the program 1005 would not need to be installed again because it has already been installed when computer 1002a was accessed.

Figure 54 is flow chart illustrating a different embodiment of the present invention. In this system, when the computer 1001 encounters an object on computer 1002a, it identifies at 1040 each program embedded within the object. It then retrieves one or more programs over the network, and then installs them at the client computer 1001, but without the use of a persistent storage mechanism. Thus, although the program is executed on the client computer 1001, the embedded program must be downloaded each time it is encountered because no persistent storage mechanism is used. This type of installation procedure may be more secure, and has been used in some of the early Java implementations.

A system in which software is downloaded over the network, perhaps from an untrusted server, has significant security risks associated with it, and for this reason, security restrictions may be placed on computer programs downloaded from the network. Thus, a downloaded computer

program may be unable access some of the resources of a client computer or of the network generally. In some embodiments, however, a downloaded program may be tested for authenticity and safety through a code signing procedure, or through a code verifying procedure. If such a program passes such authenticity tests, it may be given more complete access to system or network resources.

Figure 55 is a diagram of the relationships between six objects or documents 5301-5306. The six documents are linked to each other in the manner shown and hereinafter described. Document 5301 contains three links (5310, 5312, and 5314); one to each of the documents 5302, 5303, and 5304. Document 5302 contains two links, one link 5316 to document 5305, and another link 5318 to document 5306. Document 5305 contains a link 5320 back to document 5302, and document 5306 contains a link 5322 to document 5304. Each of these documents is stored on a server within a network, and may incorporate or have embedded within it objects stored on other servers. The documents 5301-5306 may be stored on the same server, or may be stored on various computers distributed throughout the network.

Figure 56a shows a representation of a video display screen 5404 for a computer such as that of Figure 1. The area 5404 represents the area on a screen within which images, text, video, and other type of data or multimedia objects can be displayed and manipulated. On the display 5404 shown in Figure 56a, a number of icons or objects 5402 are arranged, along with another type of object, window 5406. The window 5406 is a representation of a document retrieval, browsing, and/or viewing program that is used to view information either stored locally on the computer or retrieved over a network. The window 5406 has a title area 5408 that displays the title of the document being displayed. The title area 5408 could also display the location or address of the document being displayed, or also the universal resource locator of the document being displayed. Alternatively, an additional area within the window could be used for displaying the universal resource locator. The contents of the document are shown in displayed within the window 5406 in Figure 56a, but it should be understood that the contents could be displayed in other ways. For example, the contents could be displayed on the entire desktop, or a portion of the desktop. In another embodiment, the contents might be scrolled on the screen, perhaps under other windows.

Shown within the document viewing area of the window 5406 in Figure 56a is the contents of the document 5301 from Figure 55. The document 5301 has been displayed in the window 5406

in response to a user query, which might involve a key word search or might involve the user specifying the address or resource locator of document 5301. The document 5301 could be also be displayed within the window 5406 in response to the selection of a link in another document (not shown) that points to the document 5301.

The links 5310, 5312, and 5314 are visible within the document 5301 shown in Figure 56a. The link 5310 is actually a banner graphic advertisement for Gaylord's Restaurant, and it also acts as the link 5310. Selection of the link 5310 by an input device (e.g., mouse or voice recognition) causes the home page for Gaylord's Restaurant to be retrieved from the network and displayed within the window 5406. The links 5312 and 5314 are hypertext links to two other documents. Selection of one of these links with the input device causes the document corresponding to the selected link to be displayed in the window 5406. In an alternate embodiment, rather than displaying the selected document in the window 5406 when the link is selected, the selected document could be displayed elsewhere on the display 5404, such as in a new window.

Figures 56b through 56f show the contents of documents 5302 through 5306 rendered in the window 5406. In Figure 56b, for example, the contents of document 5302 have been rendered in the window 5406, and the title bar 5408 displays the title of the document 5302. The contents of document 5302 may be displayed in the window 5406 in response to one of any number of user requests. One such user request might be simply a query by the user to display the document 5302, where the user specifies for the viewing program the address or the resource locator for the document 5302. Another request might arise when the user has the contents of document 5301 displayed in the window 5406, as shown in Figure 56a. If link 5310 is selected in Figure 56a, the viewing program retrieves the document 5302 and then displays it in the window 5406.

Figure 56c shows the document 5303 displayed in window 5406. This document can be displayed in response to the user selecting link 5312 in Figure 56a. Similarly, the document 5304 can be displayed as shown in Figure 56d in response to the user selecting link 5314 in Figure 56a.

In Figure 56b, hypertext links 5316 and 5318 point to documents 5305 and 5306, respectively. When the user selects link 5316, document 5305 is retrieved and rendered in the window 5406 as shown in Figure 56e. And when the user selects link 5318, document 5306 is retrieved over the network and rendered in the window 5406 as shown in Figure 56f.

Figure 57 shows a representation of four documents that might correspond to search documents within a database that satisfy a query formulated by the user. In the classic case, and in Figure 57, a "document" is a textual representation of an article, book, or other literary work. But a "document" need not only be text--it could also be a picture, video clip, or information in another format, such as might be available in a multimedia encyclopedia.

Referring again to Figure 57, document 100a is five "views" in length. Each "view" corresponds to that amount of information that can be shown on the monitor at any one time. The number of views in a search document therefore depends on the size of the monitor's display. The five views that compose document 100a are shown as views 101a, 102a, 103a, 104a, and 105a. Document 200a is six views in length (201a, 202a, 203a, 204a, 205a, and 206a). Document 300a is three views in length (301a, 302a, and 303a) and document 400a is five views in length (401a, 402a, 403a, 404a, and 405a).

The search documents of Figure 57 might represent the search documents found as a result of a query formulated to find all the documents in the database that include the phrase "Hadley v. Baxendale." Each X in the search documents represents an occurrence of the phrase "Hadley v. Baxendale." As can be seen, the phrase "Hadley v. Baxendale" can be found in search document 1 at two separate locations. Document 200a has six occurrences, and search document 300a has three. Search document 400a has one occurrence--the title of search document 400a is "Hadley v. Baxendale."

There are also "related documents" (500a, 600a, and 700a) shown in Figure 57. A related document is a document that is somehow explicitly associated, linked, or otherwise connected to one of the search documents. For example, if search document 1 (100a) is a judicial opinion, a related document might be a subsequent opinion in the same case (e.g., an decision on appeal). Other related documents might be an opinion or scholarly article that cites or discusses search document 100a, or a list of judicial opinions that cite the search document. Any document that is usefully associated with the search document can be considered a related document. Often, the related document does not satisfy the query, so it is usually not one of the search documents. In some circumstances, however, the related document might satisfy the query, so it can be a search document.

Related documents may also be related only to a particular view within a search document. For example, a search document that is a judicial opinion may have numerous other judicial opinions cited in the text of the opinion. These cited opinions may be "related documents," but often they relate only to a particular view within the document. Depending on the implementation of the database system, they might not be considered to be "related" to the search document as a whole. Thus, they are available as related documents only when the corresponding cite is within the currently displayed view. In such an implementation, the related documents are dependent on the view shown on the monitor at any given time.

Figure 58 shows the representation of the four search documents that satisfy the user's query. The search documents are ordered by an ordering characteristic, such as the date of publication. Other ordering characteristics can be used as appropriate for a given situation (e.g., number of query terms in a document, statistical relevance of the documents type of document, etc.). Any ordering characteristic that permits the search documents to be distinguished from one another can be appropriate. In the example of Figure 58, search document 100a is the first search document according to the ordering characteristic, and view 101a (shaded) in search document 100a is the display view shown on the monitor. (The view shown on the monitor at any given time is the "display view.") Once view 101a is displayed on the monitor, the user reads, studies or otherwise observes the displayed information. When the user wishes to change the display view, he or she uses the input device to cause the system to display either (a) a different view in the search document 100a, or (b) a view from one of the other documents 200a, 300a, 400a, 500a, 600a, or 700a.

The user uses one or more input devices to request particular views. For example, an input device might be a keyboard that includes a "next page" key and a "next document" key. The "next page" key requests the next successive view (view 102a) within the document currently being viewed (document 100a). The "next document" view requests the first view (view 201a) of the next successive search document according to the ordering characteristic (document 200a). Many database systems have "next page" and "next document" commands or keys (e.g., Westlaw, LEXIS/NEXIS, and West Publishing Company's CD-ROM products), as well as others (e.g., "previous document," "previous page"). Westlaw also permits a user to request a particular search document or "page" by typing a command. For example, to view search document three (300a),

the user types "r3"; to request page 2 (i.e., view 2) within the currently displayed document, the user types "p2." And in some systems, multiple commands can be executed together by separating them with a semicolon, so page two from document three (view 302a) can be requested with a single command: "r3;p2."

In the systems of the prior art, when the database system receives the command to display a different view, the requested view must be loaded from the database 30 before it can be displayed on the monitor 27. Since retrieving information from the database is time-consuming, this loading process is undesirably slow. But in a system employing the present invention, the time required to respond to the user's request for a different view (the "requested view") is reduced by taking advantage of the fact that it is often possible to predict the requested view before the user actually requests it. In the present invention, the view(s) that the user is likely to next request are preloaded while the user is reading the displayed view.

Thus, in one embodiment of the present invention, the view or views (i.e., anticipated view(s)) that are likely to be next requested by the user are "preloaded" (e.g., in the background) to the extent permitted by the time the user spends reading or studying the display view. When the user does request that a different view be displayed (i.e., the user requests a "requested view"), the requested view can be very quickly displayed on the monitor if it has already been preloaded into memory. Thus, if the requested view is one of the anticipated views, the database system is able to quickly respond to the user's request for the requested view.

As shown in Figure 58, while the user is reading or studying the display view 101a, view 201a is identified as an anticipated view (signified by the arrow from view 101a to view 201a). View 201a is likely to be requested by the user because it is the first view of the "next" search document (as defined by the ordering characteristic) following search document 100a. And while the display view 101a is being viewed by the user, the database system will preload view 201a from the database into memory, before it is actually requested by the user. After view 201a is preloaded into memory, the input device is checked to see if the user has requested that another view be displayed. If the user has requested that a requested view be displayed, the database system checks to see if the requested view has been loaded into memory (e.g., as the preloaded anticipated view). If the requested view is view 201a, it will have been loaded into memory as the anticipated view, so view 201a is retrieved from memory and displayed on the monitor. Since loading the requested

view from memory is much faster than loading the requested view from the database, the time required to respond to the user's request for the requested view is shortened dramatically. If the requested view is not in memory, however, it must be retrieved from the database.

Instead of loading the entire anticipated view before checking the input device, in other embodiments of the present invention the input device is monitored during the time the anticipated view is being preloaded into the database. If the user requests a requested view, the preloading of the anticipated view stops and the user's request is serviced. This ensures that the system is very responsive to the user's input. Such an embodiment can be implemented by checking the input device each time a segment (i.e., a portion) of the anticipated view is preloaded. If the computer is running multitasking and/or multithreading operating system, such an embodiment can alternatively be carried out using the various techniques appropriate for such an operating system.

Figure 59(a) shows a situation where view 101a (shaded) is the display view, and the retrieval system has identified four views 102a, 501a, 201a, and 401a as anticipated views. View 102a is likely to be requested by the user when the display view is view 101a because it is the next view in the document that the user is currently viewing. View 501a is a candidate for the requested view because it is the first view from a document (500a) that relates to the search document (100a) that the user is currently viewing. View 401a is also an anticipated view because the user might wish to view the document that represents the opposite extreme of the ordering characteristic (e.g., the oldest document). And as described above, view 201a is also likely to be requested by the user.

In the embodiment of Figure 59(a), the retrieval system will attempt to load as many of these anticipated views as possible while the user is studying the display view 101a. If enough time passes before the user requests a requested view, the retrieval system may preload all four of the anticipated views, thereby enhancing the likelihood that the next requested view will be in memory.

Once the user issues a request for a requested view, the requested view is loaded from memory (or from the database, if necessary) and displayed on the monitor. The process of determining and preloading anticipated views then starts over. For example, if the requested view is view 201a, the display view will then become view 201a (shaded) as shown in Figure 59(b). The anticipated views would also change, and might be identified as indicated by the arrows.

Figure 60(a) shows another representation of four search documents showing term views 111, 112a, 211, 212, 213, 214, 311a, 312a, and 411. In Figure 60(a), a term view is a view that

has at least one search term from the query. And as can be seen from document 100a in Figure 60(a), the boundaries of these term views may or may not correspond to the boundaries of views 101a, 102a, 103a, and 104a. Term views may also be anticipated views because the user might request as a requested view the next view having one or more of the terms in the query. Some systems provide a command for this purpose (e.g., in Westlaw, the command is "t").

Figure 60(b) shows the representation of the four search documents showing other term views 171, 271, 272, 371, and 471. These term views are made up of a small number of words surrounding each occurrence of a search term in the search documents. Since the number of words surrounding the search terms is small, more than one set of words can fit on the screen at a given time. Thus, the term view in this embodiment includes information from different parts of the document. The "KWIC" display format in the LEXIS/NEXIS system operates similarly.

Figure 61 shows another representation of the four search documents showing subdocument views 121, 122, 131, 141, 221, 231, 232, 233, 321, 331, 421, 431, and 441. The subdocuments are shown in Figure 61 as 120, 130, 140, 220, 230, 240, 320, 330, 420, 430 and 440. A subdocument is any logically separable or identifiable portion of a document. For example, if a document is a judicial opinion, there might be subdocuments for the title and citation for the case, for each of the headnotes, for the opinion itself, and for any dissenting opinions. A subdocument view is a view within a subdocument.

Subdocument views may be anticipated views because the user often is particularly interested in a particular portion of the search documents. If the search documents consist of a series of judicial opinions, for example, a user may only wish to view, for each of the search documents, the subdocument for the majority opinion (and not the headnotes, dissenting opinions, etc.). Thus, it may be appropriate for the anticipated views to be drawn primarily from a particular type of subdocument.

In other situations, however, the user may only wish to see the first subdocument view for each subdocument. It would be appropriate in these situations for the anticipated views to be primarily the first views from the various subdocuments within each document.

The retrieval system of the present invention identifies anticipated documents by focussing on the current display view. The current display view gives clues as to which view might be requested by the user because the display view identifies the user's progress in browsing the search

documents. In other words, the current display view identifies which search document in the sequence of search documents is currently being viewed. This information is useful because the search document immediately following and preceding the current search document (as defined by the ordering characteristic) is often the search document next requested by the user.

The view displayed just prior to the displayed view might also be a consideration in determining the anticipated views if it tends to show a pattern that can identify the user's next requested view. For example, referring to Figure 61, if the user requests view 131 of search document 100a, and then requests view 231 of search document 200a, the retrieval system can consider these two consecutive display views and determine that an appropriate anticipated view is view 331 of search document 300a. View 331 is the first view of subdocument 330, which could be of the same type as subdocuments 130 and 230, the two subdocuments previously viewed by the user. Since the goal is to accurately predict the next requested view, considering the views that the user requested in the past may be helpful if it tends to identify the views that the user will request in the future.

In general, any appropriate adaptive prediction scheme can be used that uses the user's history of requested views (and display views) to accurately determine which views are likely to be next requested by the user. It might be appropriate in some cases to consider many display views in determining appropriate anticipated views. Longer histories may tend to identify patterns that would not show up if only a small number of recent display views are considered.

Tendencies can even be monitored over more than one research session in situations where a particular user or group of users tend to request views in a particular pattern each time research is done. In addition, the user could be prompted to indicate the type of research being undertaken, which may give clues as to what type of anticipated views are appropriate for efficient operation. Finally, the particular databases used or type of research being done can be monitored by the database system and advantageously taken into account in determining anticipated views.

In the preferred embodiment of the present invention, the anticipated views are drawn from both related documents and search documents. A fundamental distinction between related documents and search documents is that related documents are statically-related to the search documents, whereas search documents are dynamically-related to one another. This difference is significant because unlike statically-related documents, no predefined link needs to be set up for

search documents. A statically-related document is always associated with a particular document, regardless of the query (the related document is therefore statically-related). The search documents, on the other hand, are related to each other by the query. Since the query changes with each search, the search documents are considered dynamically-related to one another.

Some of the recent CD-ROM products have implemented features such as hyperlinked text, and timeline-linked text (clicking on a time-line item will take the user to a relevant article). See The Top 100 CD-ROMs, PC Magazine, Sep. 13, 1994, p. 115. Links of this nature are static because they always apply and do not depend on any particular query run by the user.

The search documents are ordered by an ordering characteristic as described previously. Thus, when a "next document" is requested, it is assumed that the search document requested by a "next document" command is the search document that is "next" according to the ordering characteristic. If the search documents are ordered by publication date, for example, the "next document" will be interpreted as a request for the search document with the next oldest publication date.

In one embodiment of the present invention, it is possible to make a number of different ordering characteristics available for use by the user in browsing the search documents. For example, Figure 62 shows seven documents labelled "a" through "g" ordered according to four different ordering characteristics. When the display view is in document "a," the "next document" command can be a request for four different documents (i.e., "b," "e," "f," or "c"), depending on the particular ordering characteristic used. More than one ordering characteristic must therefore be considered when determining anticipated views if the user is capable of moving to a "next document" in the context of more than one ordering characteristic. This feature can be enabled by an input device command that allows the user to select the desired ordering characteristic.

The present invention is applicable to single-user, multiple-user, and many-user databases, but the present invention is most effective when used in connection with single-user databases. The efficient operation of the invention depends on being able to retrieve data from the database very frequently, perhaps continually. The present invention is quite effective with single-user databases such as those on CD-ROM or other mass storage devices (this might also include a hard drive implementation). In a single-user database, the no other demands are being made on the database by other users, so the database is often idle.

But since a many-user or multiple-user database must be shared among more than one user, such a database will often be receiving simultaneous and continual requests for data. Databases in such a system are rarely idle, so there is little time to preload anticipated views into memory. In such a situation, the present invention will not be as effective in improving the response time to users' requests for requested views. But in many-user or multiple-user database systems where the database is not as busy, the present invention can be effective in reducing response times to users' requests for information.

Figure 63 is a flow chart of the operation of the database system in one embodiment of the present invention. A system in one embodiment of the present invention begins by executing a query to identify the search documents. This step is carried out by search logic 51. The remaining steps shown in Figure 63 (described below) are carried out by retrieval logic 52. Both the search logic 51 and the retrieval logic 52 are part of the program logic, which is often software. As one skilled in the art will recognize, in a software implementation the search logic 52 and the retrieval logic 52 may or may not be integral or intertwined parts of the same computer program (i.e., program logic).

As dictated by the retrieval logic 52, the database system then loads into memory a view from one of the search documents. See Figure 63. This first display view is then displayed on the monitor. Normally the user will take a few moments to read or study the display view. During this time, one or more anticipated views are identified. The anticipated views are views that the user is likely to request be displayed on the monitor after the display view.

The database system then begins to preload these anticipated views into memory from the database, while also continually monitoring the input device to determine if the user has issued a request to display a different view (i.e., a "requested view") on the monitor. Anticipated views are loaded into memory until the user requests a requested view.

When the user does makes such a request, the database system then determines whether the requested view is in memory. The requested view may be in memory because it could have been preloaded into memory as an anticipated view. If the requested view is in memory, the requested view becomes the new display view, and it is displayed on the monitor. But if the requested view is not in memory, the requested view must first be loaded from the database before it can be displayed on the monitor as the display view.

The anticipated views are a function of the display view because the views that the user is likely to request depend to some degree on the view the user is currently reading. In other words, those views that are anticipated views when view 101a is the display view are not likely to be the same as the anticipated views when view 202a is the display view. Therefore, as shown in Figure 63, the anticipated views are determined each time the display view changes.

When the display view is changed, the anticipated views for the prior display view can remain in memory so that they are available if they are ever requested by the user. But if memory is limited, the anticipated views for the prior display view can be deleted from memory, preferably in an efficient manner (e.g., anticipated views common to both the new display view and the prior display view are not deleted from memory). It is best to delete those views that are not likely to be requested by the user. It may also be appropriate to consider whether a view is likely to become an anticipated view in the future.

Figure 64 shows a flow chart representing another embodiment of the present invention where anticipated views from prior display views are deleted if memory is full. The views deleted are those that are not anticipated views for the new display view. This will presumably make room for new anticipated views to be preloaded into memory (if not all of the anticipated views are already in memory).

The number of anticipated views for a given display view does not have to be a predetermined or constant number, but rather can vary depending on memory available. Typically, the number of anticipated views for a display view is a trade-off between the amount of memory available and the desired speed of retrieval. In instances where memory is plentiful, where the number of search documents is few, and/or where the search documents are small, it may be possible for all of the search documents to be completely loaded into memory. In such a situation, the number of anticipated views for a given display view could be as high as the total number of views in the search documents. At the other end of the spectrum, there might be only one or two anticipated views for each display view if memory is limited.

Embodiments of the present invention can vary as to how anticipated views are preloaded into memory. In the embodiments of Figures 63 and 64, one anticipated view at a time is preloaded into memory, and the retrieval system does not begin preloading a second anticipated view into

memory until the prior anticipated view is completely preloaded into memory. In other embodiments, anticipated views are simultaneously preloaded.

Simultaneous preloading of multiple anticipated views can be done in a number of ways. In a multitasking operating system, for example, an appropriate time-slicing procedure can be used to preload the anticipated views so that they are preloaded simultaneously. In another embodiment, one segment from each anticipated view is preloaded in turn, and the cycle is repeated until all the anticipated views are fully preloaded into memory (or until the user's request for a requested view interrupts the preloading process). A segment is any portion of an anticipated view, such as one or two lines or even a single byte of the anticipated view.

Figure 65 shows a simple implementation of the simultaneous preload concept, where the database system preloads a segment of a first anticipated view into memory, and then preloads a segment of a second anticipated view into memory. These steps continue until either the user requests a requested view, or both anticipated views are fully preloaded into memory. When the user requests a requested view, the database system checks to see if that requested view is in memory. If the requested view is only partially preloaded into memory, that portion in memory can be written to the monitor and the remaining portion loaded from the database. The response time in this situation will still be better than if the entire requested view has to be loaded from the database. The present invention has been principally described in the context of accessing the database and identifying search documents through a search term query. The present invention can be applicable in other research-related contexts where search documents are identified using another type of entry path. For example, a time-line can be used for locating information or documents that are associated with a given time or time-frame. Another information access method uses a topic tree that permits a user to choose from successively narrowing topics until the desired topic is located. It is possible for the present invention to be applicable even in other non-research contexts where similar preloading techniques may permit efficient navigation of information and/or short response times. The present invention can also be used in combination with caching systems where previously-displayed views are stored for repeated use.

The present invention has been primarily described in the context of a general purpose computer implementation. As one skilled in the art will recognize, however, it is possible to construct a specialized machine that can carry out the present invention.

Generally, although the present invention has been shown and described with respect to specific embodiments and/or implementations, various changes and modifications, even if not shown or specifically described herein, may be applicable to the present invention, and are deemed to lie within the spirit and scope of the present invention as defined by the appended claims. Any specific features or aspects of the embodiments or implementations described or illustrated herein are not intended to limit the present invention in a manner not required by the appended claims.

INCORPORATION BY REFERENCE

The following pending U.S. patent applications are hereby fully incorporated by reference into this application: application serial nos. 08/474,921 (entitled "Document Retrieval System Employing a Preloading Procedure"), filed June 7, 1995, now U.S. Patent No. ____; 08/487,925 (entitled "Document Research System and Method for Displaying Citing Documents"), filed June 7, 1995; 60/028,251 (entitled "A Method and System for Communicating Information About the Contents of a Document"), filed October 8, 1996; 60/047,554 ("A Document Retrieval System Including the Use of Profile Information"), filed May 22, 1997; 60/052,830 ("A System and Method for Communicating Information Relating to a Network Resource"), filed July 17, 1997; and 08/936,910 (entitled "A System and Method of Communicating Information Relating to a Network Resource"), filed September 25, 1997.

In addition, the following publications listed below are also hereby fully incorporated by reference to the extent that they enable, provide support for, provide a background for, or teach methodology, techniques, and/or procedures employed herein.

Reference 1: Yellin, The Java Application Programming Interface: Volumes 1 & 2 (Addison Wesley 1996)

Reference 2: Campione, The Java Tutorial (Addison Wesley 1996)

Reference 3: Chan, The Java Class Libraries (Addison Wesley 1997)

Reference 4: Lemay et al., Java in 21 Days (Sams.Net 1996)

Reference 5: Sun Microsystems, Inc., The JavaBeans 1.0 API Specification (Sun Microsystems 1996) (available at <http://java.sun.com/beans>)

Reference 6: Sun Microsystems, Inc., The Java 1.1 API Specification (Sun Microsystems 1997) (available at <http://java.sun.com/>)

Reference 7: Bell, "Make Java fast: Optimize!," JavaWorld April 1997 (JavaWorld 1997) (available at <http://www.javaworld.com/>)

Reference 8: Vanhelsuwe, "How to make Java applets start faster," JavaWorld December 1996 (JavaWorld 1996) (available at <http://www.javaworld.com/>)

Reference 9: Danesh, JavaScript in a Week (Sams.Net 1996)

Reference 10: Graham, HTML Sourcebook (2d ed. John Wiley & Sons 1996)

Reference 11: Tanenbaum, Computer Networks (2d ed. Prentice Hall 1989)

Reference 12: Jamsa, Internet Programming (Jamsa Press 1995)

Reference 13: Comer, Internetworking with TCP/IP, Volumes 1, 2, & 3 (3d ed. Prentice Hall 1995)

Reference 14: Lemay, Official Marimba Guide to Castanet (Sams.Net 1997)

Reference 15: Kovel et al., The Lotus Notes Idea Book (Addison Wesley 1996)

Reference 16: Schneier, Applied Cryptography (2nd ed. John Wiley & Sons 1996)

Reference 17: Adkins, Internet Security Professional Reference (New Riders 1996)

Reference 18: Chappell, Understanding ActiveX and OLE (Microsoft Press 1996)

Reference 19: Denning, OLE Controls Inside Out (Microsoft 1995)

Reference 20: Brockschmidt, Inside OLE (2d ed. Microsoft 1995)

Reference 21: Siegel, CORBA Fundamentals and Programming (John Wiley & Sons 1996)

Reference 22: Petzold, Programming Windows 95 (Microsoft 1996)

Reference 23: Proise, Programming Windows 95 with MFC (Microsoft Press 1996)

Reference 24: Microsoft Corporation, Windows NT Server Resource Kit (Microsoft Press 1996)

Reference 25: Russel, Running Windows NT Server (Microsoft Press 1997)

Reference 26: Chapman, Building Internet Applications with Delphi 2 (Que 1996)

Reference 27: Cantu, Mastering Delphi 3 (Sybex 1997)